Augmented Semantic Explanations for Collaborative Filtering Recommendations

Mohammed Alshammari[1,2] and Olfa Nasraoui[1]
[1]*Knowledge Discovery and Web Mining Lab, CECS Department, University of Louisville, Louisville, Kentucky 40292, U.S.A.*
[2]*Northern Border University, Rafha 76313, Saudi Arabia*

Keywords: Recommender Systems, Semantic Web, Collaborative Filtering, Matrix Factorization.

Abstract: Collaborative Filtering techniques provide the ability to handle big and sparse data to predict the rating for unseen items with high accuracy. However, they fail to justify their output. The main objective of this paper is to present a novel approach that employs Semantic Web technologies to generate explanations for the output of black box recommender systems. The proposed model significantly outperforms state-of-the-art baseline models in terms of the error rate. Moreover, it produces more explainable items than all baseline approaches.

1 INTRODUCTION

Matrix factorization (MF) Koren et al. (2009) is a powerful collaborative filtering technique. However, MF lacks transparency even though it produces accurate recommendations. This means that despite its efficient handling of big data and high accuracy in predicting unseen items' ratings, it fails to justify its output. Thus, it is called a black box recommender system. Moreover, users' explicit preferences may not be enough for the model to consider some items in the process of recommending new items. Since users may not have given new items any preferences, these items may be discarded. This cold-start problem is well-known in the recommender systems field.

Extra information can be used to overcome both the black box and cold-start problems. Information can be found in semantic KGs built using semantic web technologies. Linked open data (LOD) Bizer et al. (2009) is a platform for linked, structured, and connected data on the web. The goal of LOD is to make information machine processable and semantically linked. For example, in the movie domain, information about movie stars or directors is available in a linked way. If an actor has starred in two movies, those two movies are linked. This can help us infer new facts about movies that eventually lead to the resolution of the cold start and transparency problems mentioned earlier.

Our research question is as follows: can we build semantic knowledge graphs (KGs) about users, items, and attributes to generate explanations for a black box recommender system, while maintaining high prediction accuracy?

This paper's contribution consists of solving the problem of a non-transparent MF recommender system, in addition to constructing semantic KGs about users, items, and attributes for the inference and explanation process.

2 RELATED WORK

Explaining black box recommender systems has been the subject of several studies. RippleNet Wang et al. (2018) is an approach that used KGs in collaborative filtering to provide side information for the system in order to overcome sparsity and the cold-start problem. This black box system takes advantage of KGs, which are constructed using Microsoft Satori, to better enhance recommendation accuracy and transparency. The authors simulate the idea of water ripple propagation in understanding user preferences by iteratively considering more side information and propagating the user interests. In the evaluation section, the authors claim that their model is better than state-of-the-art models. The research of Ai et al. (2018) focuses on adding explanations to a black box recommender system by using structured knowledge bases. The system takes advantage of historical user preferences to produce accurate recommendations and structured knowledge bases about users and items for generating

Alshammari, M. and Nasraoui, O.
Augmented Semantic Explanations for Collaborative Filtering Recommendations.
DOI: 10.5220/0008070900830088
In *Proceedings of the 11th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2019)*, pages 83-88
ISBN: 978-989-758-382-7

justifications. After the model recommends items, a soft matching algorithm is used, utilizing the knowledge bases to provide personalized explanations for the recommendations. The authors argue that their model outperforms other baseline methods. Bellini et al. (2018) focuses on the issue of explaining the output of a black box recommender system. In that work, the SemAuto recommender system is built using the autoencoder neural network technique, which is aware of KGs retrieved from the semantic web. The KGs are adopted for explanation generation. The authors claim that explanations increase the users' satisfaction, loyalty, and trust in the system. In their study, three explanation styles are proposed: popularity-based, pointwise personalized, and pairwise personalized. For evaluation, an A/B test was conducted to measure the transparency of, trust in, satisfaction with, persuasiveness of, and effectiveness of the proposed explanations. The pairwise method was preferred by most users over the pointwise method. Abdollahi and Nasraoui (2017) investigates the possibility of generating explanations for the output of a black box system using a neighborhood technique based on cosine similarity. The results show that Explainable Matrix Factorization (EMF) performs better than the baseline approaches in terms of the error rate and the explainability of the recommended items.

3 PROPOSED METHOD

3.1 Semantic Knowledge Graphs (KGs)

The web is abundant with information that is being harvested and structured into KGs. KGs are extensive networks of objects, along with their properties, their semantic types, and the relationships between objects representing factual information in a specific domain Nickel et al. (2016). Examples of KGs are DBpedia Auer et al. (2007), Freebase Bollacker et al. (2008), Wikidata Vrandečić (2012), YAGO Suchanek et al. (2007), NELL Carlson et al. (2010), and the Google Knowledge Graph Singhal (2012). In this study, DBpedia is used to build the desired KGs about users, items, and attributes. In contrast with Alshammari et al. (2018), where only one attribute (actors) was considered in building the KG and, hence, the model, more influential attributes (subject(s), actor(s), director(s), producer(s), and writer(s)) are included to find the similarity between items. The LDSD algorithm Passant (2010) is used to weigh the similarity between items. Then, Matrix Factorization (MF), Koren et al. (2009) with the added regularization term in Joint MF (JMF) Shi et al. (2013), is used for building the model.

3.2 Linked Data Semantic Distance Matrix Factorization (LDSD-MF)

The loss function of the proposed technique, Linked Data Semantic Distance Matrix Factorization (LDSD-MF), is inspired by the work of Koren et al. (2009) and Shi et al. (2013) as follows:

$$J = \sum_{u,i\in R}(R_{u,i} - p_u q_i^T)^2 + \frac{\gamma}{2}\sum_{i,j\in S^{ldsd}}(S^{ldsd}_{i,j} - q_i q_j^T)^2 + \frac{\beta}{2}(\| p_u \|^2 + \| q_i \|^2). \quad (1)$$

$R_{u,i}$ represents the rating for item i by user u. p_u and q_i represent the low dimensional latent space of users and items, respectively. S^{ldsd} is the semantic KG. q_i and q_j indicate two items in S^{ldsd}, and γ is a coefficient that weighs the contribution of the new term, S^{ldsd}. Stochastic gradient descent Funk (2006) is employed to update p and q iteratively until J converges.

The updating rules are given by:

$$p_u^{(t+1)} \leftarrow p_u^{(t)} + \alpha(2(R_{u,i} - p_u^{(t)}(q_i^{(t)})^T)q_i^{(t)} - \beta p_u^{(t)}), \quad (2)$$

$$q_i^{(t+1)} \leftarrow q_i^{(t)} + \alpha(2(R_{u,i} - p_u^{(t)}(q_i^{(t)})^T)p_u^{(t)} + 2\gamma(S^{ldsd}_{i,j} - q_i^{(t)}(q_j^{(t)})^T)q_j^{(t)} - \beta q_i^{(t)}). \quad (3)$$

The KG is constructed using an approach following Alshammari et al. (2018). In addition to the known rating used to update q_i, the KG also contributes to the final predicted rating of item i by user u.

4 EXPERIMENTAL EVALUATION

In this study, the MovieLens 100K benchmark dataset is used. The total number of users is 943, and that of movies is 1,862. SPARQL, a semantic web query language, is used for the mapping process between MovieLens and DBpedia, and movie titles are used for the mapping. The results indicate that 1,012 movies intersected in the two datasets. The reasons for this reduction are either absent movies in DBpedia or different spellings. The mapping also resulted in a decrease in the total number of ratings to 60K. All ratings are normalized to 1, and the hyper-parameters are set to α = 0.01, β = 0.1, and γ = 0.9, after being tuned using cross-validation. 90% of the ratings are used for training the model, and 10% are used for testing the model. Since our method randomly initializes the user and item latent spaces, an average of 10 experiments is reported.

Table 1: Numeric values of selected attributes in the experiment, with unique IDs in the second column, the total number of triples for movies in the third column, and the total number of triples for users in the fourth column.

Attribute	Unique ID	Triple (movies)	Triple (users)
Subject	4996	19983	818784
Actor	4165	6770	332484
Director	1193	1577	92008
Producer	1154	1868	103943
Writer	1491	1944	110692

Five different properties are extracted from the semantic KG DBpedia: subject, actor, director, producer, and writer. The total number of unique subjects is shown in the second column of Table 1. The third column in Table 1 shows the total number of previously existing triples of movies and attributes in DBpedia. An example could be ”Mel Gibson is starring in Braveheart.” The fourth column in Table 1 describes the size of the constructed semantic KG with the total number of triples in each KG. For example, ”User 581 likes the actor Ben Kingsley to a certain degree.”

Five baseline methods are used for comparison: MF Koren et al. (2009), EMF Abdollahi and Nasraoui (2016) Abdollahi and Nasraoui (2017) Abdollahi (2017), Probabilistic Matrix Factorization (PMF) Salakhutdinov and Mnih (2007), Asymmetric Matrix Factorization (AMF) BenAbdallah et al. (2010), and Asymmetric Semantic Explainable Matrix Factorization (ASEMF_UIB) Alshammari et al. (2018).

Several metrics are used to evaluate the recommender system. The first metric is the error rate in equation (4), while the remaining metrics are the Mean Explainability Precision (MEP), Mean Explainability Recall (MER), and the harmonic mean of the precision and recall (xF-score) Abdollahi and Nasraoui (2017), in equations (5-7).

$$RMSE = \sqrt{\frac{1}{\mid T \mid} \sum_{(u,i) \in T} \left(r'_{ui} - r_{ui} \right)^2}. \qquad (4)$$

T represents the total number of predictions, r'_{ui} represents the predicted rating on item i by user u, and r_{ui} is the actual rating on item i by user u.

$$MEP = \frac{1}{|U|} \sum_{u \in U} \frac{|\mathcal{R} \cap W|}{|\mathcal{R}|}. \qquad (5)$$

$$MER = \frac{1}{|U|} \sum_{u \in U} \frac{|\mathcal{R} \cap W|}{|W|}. \qquad (6)$$

$$xF - score = 2 * \frac{MEP * MER}{MEP + MER}. \qquad (7)$$

U represents the set of users, $\mathcal{R}$ is the set of recommended items, and W denotes the set of explainable items. MEP computes the ratio of recommended and explainable items to the total number of recommended items over all users. Similarly, MER calculates the recommended and explainable items over the total number of explainable items, again, over all users. The xF-score is the harmonic mean of MEP and MER.

Table 2: RMSE, varying the number of hidden features, K.

RMSE						
K	MF	EMF	PMF	AMF	ASEMF_UIB	LDSD-MF
10	0.205	0.205	0.698	0.236	0.205	**0.204**
20	0.212	0.211	0.698	0.27	0.204	0.204
30	0.214	0.215	0.698	0.309	0.204	0.204
40	0.216	0.217	0.7	0.344	0.203	0.205
50	0.217	0.217	0.7	0.374	0.203	0.206

Our hypothesis for the significance test is that our model is better than baseline approaches using all metrics. The null hypothesis that we are trying to reject is that the mean of all metrics for all models are equal by conducting a t-test experiments. The models are ran 10 times while randomly initializing the user and item latent factors, then we calculated all metrics and did the significance tests which are reported in this paper.

Table 3: RMSE significance test results in the movie domain (K = 10).

Model 1	Model 2	p-value
MF	**LDSDMF**	2.3e-07
EMF	**LDSDMF**	4.8e-08
PMF	**LDSDMF**	4.04e-54
AMF	**LDSDMF**	6.6e-22
ASEMF_UIB	**LDSDMF**	1.3e-07

4.1 Discussion

Table 2 shows the error rates of all the methods. The best values are in bold (the lower the value, the better). When K = 10, LDSDMF significantly outperforms all the other methods with a small p-value as shown in Table 3; however, it competes with $ASEMF_{UIB}$ as the number of hidden features increases.

In Figures 1 and 2, there are six graphs showing the performance of all models while varying θ^s and θ^n. θ^s is a threshold for items to be considered semantically explainable or not, and θ^n is a threshold for items to be explainable based on the neighborhood technique used in the baseline EMF (Abdollahi and Nasraoui, 2017). The formula for generating the neighborhood-based explainability matrix is

$$W_{ui} = \begin{cases} \frac{|N'(u)|}{|N_k(u)|} & if \quad \frac{|N'(u)|}{|N_k(u)|} > \theta^n \\ 0 & otherwise, \end{cases} \qquad (8)$$

where $N'(u)$ denotes the set of neighbors of user u

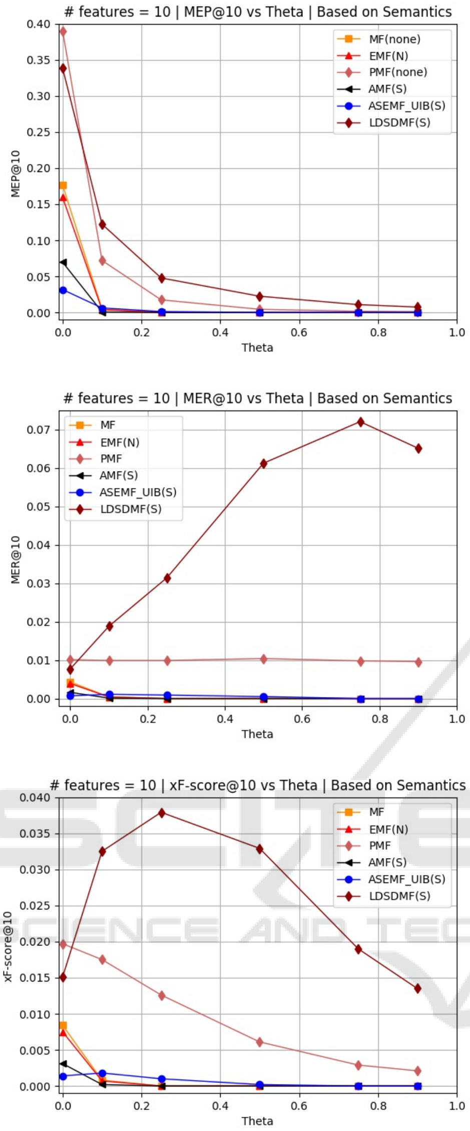

Figure 1: The upper graph shows the results of MEP@10 for all methods, while the middle one shows MER@10 for all methods, and the lower graph illustrates the results of all methods using the xF-score metric, which utilizes semantic KGs against K.

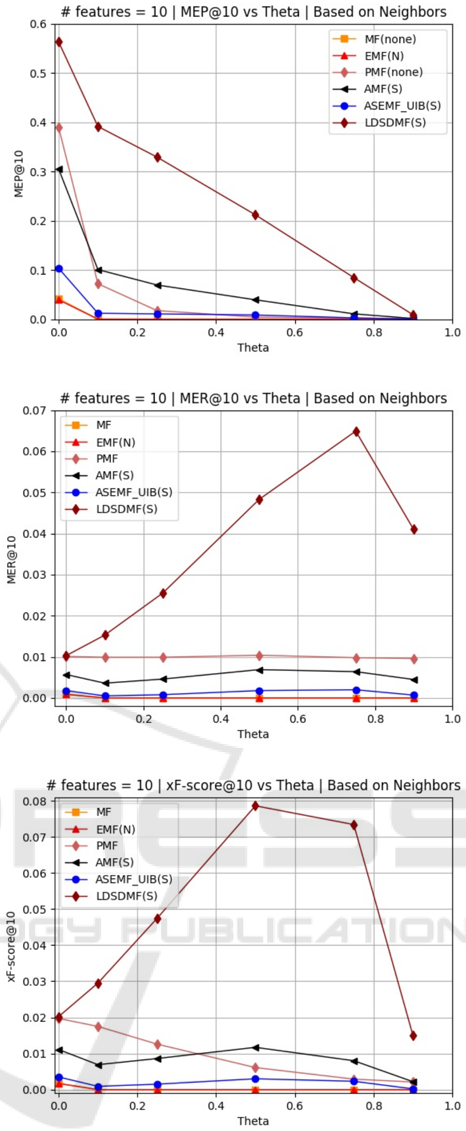

Figure 2: The upper graph shows the results of MEP@10 for all methods, while the middle one shows the MER@10 results for all methods, and the lower graph illustrates the results of all methods using the neighborhood explainability graph against K.

who rated item i, and $N_k(u)$ depicts the list of the k nearest neighbors of u.

The three graphs in Figure 1 illustrate that when θ^s is set to 0, which means that all items (even those with a small explainability value) are considered explainable, the baseline PMF is the winner. However, when adding more restrictions to items to be considered semantically explainable, the proposed method, LDSDMF, significantly outperformed the other methods for all θ^s values by all metrics (MEP, MER, and $xF-score$). Tables 4, 5, and 6 present the significance test results.

Graphs in Figure 2 present the models' performance when measuring the explainability of the recommended items based on the neighborhood technique. Our model, LDSDMF, significantly exceeded all baseline methods in all three metrics (see Tables 7, 8, and 9 for significance test results). This observation shows that our proposed method recommends more accurate explainable items, based on semantic KGs and neighborhood based techniques, than all the baseline methods.

Table 4: MEP@10 significance test results ($K = 10$ and $\theta^s = 0.25$) using semantic KGs.

Model 1	Model 2	p-value
MF	**LDSDMF**	8.06e-23
EMF	**LDSDMF**	8.1e-23
PMF	**LDSDMF**	3.05e-17
AMF	**LDSDMF**	8.06e-23
ASEMF_UIB	**LDSDMF**	2.6e-20

Table 5: MER@10 significance test results ($K = 10$ and $\theta^s = 0.25$) using semantic KGs.

Model 1	Model 2	p-value
MF	**LDSDMF**	6.2e-21
EMF	**LDSDMF**	6.3e-21
PMF	**LDSDMF**	2.1e-15
AMF	**LDSDMF**	6.2e-21
ASEMF_UIB	**LDSDMF**	1.3e-19

Table 6: xF-score@10 significance test results ($K = 10$ and $\theta^s = 0.25$) using semantic KGs.

Model 1	Model 2	p-value
MF	**LDSDMF**	1.1e-21
EMF	**LDSDMF**	1.1e-21
PMF	**LDSDMF**	5.1e-16
AMF	**LDSDMF**	1.1e-21
ASEMF_UIB	**LDSDMF**	5.6e-20

Table 7: MEP@10 significance test results ($K = 10$ and $\theta^n = 0.25$) using neighborhood technique.

Model 1	Model 2	p-value
MF	**LDSDMF**	1.9e-21
EMF	**LDSDMF**	1.9e-21
PMF	**LDSDMF**	3.9e-17
AMF	**LDSDMF**	1.2e-13
ASEMF_UIB	**LDSDMF**	9.9e-19

Table 8: MER@10 significance test results ($K = 10$ and $\theta^n = 0.25$) using neighborhood technique.

Model 1	Model 2	p-value
MF	**LDSDMF**	1.2e-21
EMF	**LDSDMF**	1.2e-21
PMF	**LDSDMF**	1.4e-15
AMF	**LDSDMF**	5.3e-15
ASEMF_UIB	**LDSDMF**	5.9e-19

4.2 Case Study

We investigated our dataset and selected a sample user as an example to show how the model captures the user's desire and recommends the next new items accordingly with an explanation. User 586 in the MovieLens dataset rated 94 movies, including *Twister (1996)* and *Tombstone (1993)* with 4-star ratings and *Apollo 13 (1995)* with a 3-star rating. All three movies are starred by Bill Paxton. *Titanic (1997)* includes the same actor in the starring actors list, and the model recommended this movie among the top 10 recommended items. Using the semantic KGs on users and attributes that were built by the model, our model succeeds in capturing the user's attribute preferences and recommends new items accordingly. Figure 3 depicts a projected example of what an explanation would look like for user 586.

Table 9: xF-score@10 significance test results ($K = 10$ and $\theta^n = 0.25$) using neighborhood technique.

Model 1	Model 2	p-value
MF	**LDSDMF**	1.1e-21
EMF	**LDSDMF**	1.1e-21
PMF	**LDSDMF**	9.2e-16
AMF	**LDSDMF**	6.4e-15
ASEMF_UIB	**LDSDMF**	5.9e-19

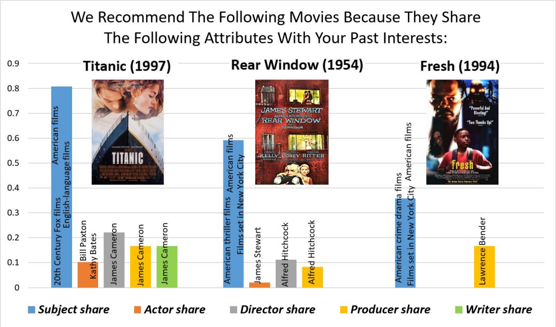

Figure 3: Example of Inferred Fact Style Explanation.

5 CONCLUSIONS

As recommendation systems become an essential component of big data and artificial intelligence (A.I.) systems, and as these systems embrace more and more sectors of society, it is becoming ever more critical to build trust and transparency into machine learning algorithms without significant loss of prediction power. Our research harnesses the power of A.I., such as KGs and semantic inference, to help build explainability into accurate black box predictive systems in a way that is modular and extensible to a variety of prediction tasks within and beyond recommender systems.

REFERENCES

Abdollahi, B. (2017). *Accurate and justifiable : new algorithms for explainable recommendations.* PhD thesis.

Abdollahi, B. and Nasraoui, O. (2016). Explainable matrix factorization for collaborative filtering. In *Proceedings of the 25th International Conference Companion on World Wide Web*. ACM Press.

Abdollahi, B. and Nasraoui, O. (2017). Using explainability for constrained matrix factorization. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 79–83, Como, Italy. ACM.

Ai, Q., Azizi, V., Chen, X., and Zhang, Y. (2018). Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms*, 11(9).

Alshammari, M., Nasraoui, O., and Abdollahi, B. (2018). A semantically aware explainable recommender system using asymmetric matrix factorization. In *Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*. SCITEPRESS - Science and Technology Publications.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). DBpedia: A nucleus for a web of open data. In *The Semantic Web*, pages 722–735. Springer Berlin Heidelberg.

Bellini, V., Schiavone, A., Di Noia, T., Ragone, A., and Di Sciascio, E. (2018). Knowledge-aware autoencoders for explainable recommender systems. In *Proceedings of the 3rd Workshop on Deep Learning for Recommender Systems*, DLRS 2018, pages 24–31, New York, NY, USA. ACM.

BenAbdallah, J., Caicedo, J. C., Gonzalez, F. A., and Nasraoui, O. (2010). Multimodal image annotation using non-negative matrix factorization. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, WI-IAT '10, pages 128–135, Washington, DC, USA. IEEE Computer Society.

Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked data - the story so far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, Vancouver, Canada. ACM.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, Jr., E. R., and Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, pages 1306–1313. AAAI Press.

Funk, S. (2006). Netflix update: Try this at home. Technical report.

Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.

Nickel, M., Murphy, K., Tresp, V., and Gabrilovich, E. (2016). A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33.

Passant, A. (2010). Measuring semantic distance on linking data and using it for resources recommendations. In *AAAI spring symposium: linked data meets artificial intelligence*, volume 77, page 123.

Salakhutdinov, R. and Mnih, A. (2007). Probabilistic matrix factorization. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, pages 1257–1264, USA. Curran Associates Inc.

Shi, Y., Larson, M., and Hanjalic, A. (2013). Mining contextual movie similarity with matrix factorization for context-aware recommendation. *ACM Trans. Intell. Syst. Technol.*, 4(1):16:1–16:19.

Singhal, A. (2012). Introducing the knowledge graph: things, not strings. Technical report, Google.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: Core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web - WWW '07*. ACM Press.

Vrandečić, D. (2012). Wikidata: a new platform for collaborative data collection. In *Proceedings of the 21st international conference companion on World Wide Web - WWW '12 Companion*. ACM Press.

Wang, H., Zhang, F., Wang, J., Zhao, M., Li, W., Xie, X., and Guo, M. (2018). Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, pages 417–426, New York, NY, USA. ACM.